



# Cluster-based ensemble learning for wind power modeling from meteorological wind data

Hao Chen

Department of Technology and Safety, Arctic Centre for Sustainable Energy, UiT The Arctic University of Norway, Tromsø, Norway

## ARTICLE INFO

### Keywords:

Wind power modeling  
Clustering  
Layered ensemble learning  
Farthest first algorithm  
Stacking  
Arctic

## ABSTRACT

Reliable and efficient power modeling from meteorological wind data is vital for optimal implementation and monitoring of wind energy, and it is important for understanding turbine control, farm operational optimization, and grid load balance. Based on the idea that similar wind conditions lead to similar wind powers; this paper constructs a modeling scheme that orderly integrates three types of ensemble learning algorithms—bagging, boosting, and stacking—and clustering approaches to achieve wind power modeling from multiple wind-based meteorological factors in a wind farm. The paper also investigates the applications of different clustering algorithms and methodologies to determine cluster numbers in the modeling. The results reveal that all ensemble models with clustering exploit the intrinsic information in wind data and thus outperform models without clustering by approximately 15% on average in modeling wind power. The model with the best-performing Farthest First clustering is computationally rapid and with an improvement of around 30% compared with the baselines. Given the diversity introduced by clustering algorithms, the power modeling performance is further boosted by about 5% by introducing stacking that fuses ensembles with varying clusters. The proposed modeling framework thus demonstrates promise by delivering efficient and robust performance on the targeted problem.

## 1. Introduction

Wind energy is one of the most commercially viable renewable energy sources, due to its natural abundance and non-reliance on fossil fuels, and has thus become integral to combating climate change. The Global Wind Energy Council estimates that 355 GW of new capacity will be installed between 2020 and 2024, with almost 71 GW of new capacity per year [1]. The rising prevalence of wind energy brings new challenges for planning electricity generation and dispatching grids because wind power varies intermittently and unpredictably with prevailing wind conditions.

Establishing an accurate power model for a wind farm based on the empirical mapping of weather data is important to understand the relationship between wind and wind power generation, which in turn is significant for the safe and stable operation of a wind farm and its economic operation [2]. It is also important for farms to have a

non-parametric power curve model that can be applied as a reference profile for the online monitoring generation process [3]. This article aims to conceptualize wind power modeling by using weather data, rather than only wind speed, to compute its corresponding wind power.

### 1.1. Related works

Driven by progress in computing affordability and capability and algorithmic advances, wind power can increasingly be modeled by physical, statistical, and hybrid methodologies. However, there is still room to improve these models [4].

A few studies have considered meteorological factors in wind power modeling. R. Liu et al. [5] inputted wind speed, wind direction, and air pressure to a power model based on multivariable phase space reconstruction—the similarity of time-series and linear regression—and demonstrated its superiority for forecasting under conditions where

*Abbreviations:* NWP, Numerical Weather Prediction; K-means, K-means clustering; EM, Expectation-Maximization clustering; FF, Farthest First clustering; Canopy, Canopy clustering; X-means, X-means clustering; SSE, Sum of Squared Errors; BIC, Bayesian Information Criterion; NMAE, Normalized Mean Absolute Error; NRMSE, Normalized Root Mean Square Error; Bagging, Bootstrap aggregating; Adaboost, Adaptive boosting; REPTREE, Reduced-Error Pruning TREE; AdaRF, Adaboost with Random Forest; LR, Linear Regression; ANN, Artificial three-layer Neural Networks; AdaDT, Adaboost Decision Tree; CoV, Coefficient of Variation; 'Cls-' AdaRF, LR, ANN, AdaDT, Two-layer stacking with four clustering methods as the first and AdaRF, LR, ANN, AdaDT as the second, respectively; NCI-AdaRF, Emphasis on AdaRF without a clustering layer (same with AdaRF); FF-AdaRF, Two-layer stacking with FF clustering and AdaRF.

*E-mail address:* [hao.chen@uit.no](mailto:hao.chen@uit.no).

<https://doi.org/10.1016/j.rser.2022.112652>

Received 23 April 2021; Received in revised form 17 May 2022; Accepted 30 May 2022

Available online 17 June 2022

1364-0321/© 2022 The Author. Published by Elsevier Ltd. This is an open access article under the CC BY license (<http://creativecommons.org/licenses/by/4.0/>).

wind power series fluctuate considerably. J. Ma et al. [6] used hourly wind speed and direction at the heights of 10 m and 100 m to establish a good performance model through multivariate empirical dynamic modeling. However, this type of research focused on mapping the relationship between weather data and wind power, without examining meteorological data themselves and their potential to improve these models.

Ensemble learning also remains a popular approach to improving modeling, since it can reduce the variance and bias of learners. D. Niu et al. [7] established a wind-speed power model with wavelet decomposition and weighted random forest optimized by the niche immune lion algorithm. The model was subsequently tested in two empirical analyses. Y. Dong et al. [8] processed input data with wavelet packet decomposition and applied a stacking ensemble by evaluating the correlation coefficient between base learners for wind power forecast modeling. The model clearly showed the ensemble edge when the base learners have high accuracy and low correlations with one another. However, these studies were primarily algorithm-oriented and rarely considered wind data's inherent characteristics in detail; meanwhile, their uses of decomposition to handle data increased the modeling time complexity.

Wind has some internal trends that can be understood through data mining approaches. There have been some studies on clustering technique applications in wind power modeling. V. Kushwah et al. [9] found that clusters of time series data showed identical trend components in wind speed data using a cluster-based statistical modeling technique, which showed better performance than other statistical ones. However, purely statistical models can suffer from underfitting problems when dealing with complex data. L. Dong et al. [10] utilized cluster analyses of the Numerical Weather Prediction (NWP) since wind power and corresponding meteorological data have the characteristic of daily similarity. This suggests that the clustering model is useful in the day-ahead modeling of wind power. K. Wang et al. [11] clustered NWP data consisting of daily wind speed, pressure, humidity, and temperature by K-means and fed the data into a deep belief network for day-ahead prediction modeling, showing that reduced volatility and sophistication in NWP data drove the outperformance. This also revealed the difficulty of tuning hyperparameters in specific modeling problems, especially in network-based models.

Fortunately, decision tree algorithms do not require the adjustment of many parameters. S. Tasnim et al. [12] proposed a K-means cluster-based ensemble regression by linear and support vector regression for wind power forecast modeling and proved its superiority, with an up to 17.94% upgrade, by comparison with no-clustering and several ensemble models in seventy Australian wind sites. The upgrade, as compared with the baseline, is further enlarged to 20.63% by employing a transfer learning approach called multi-source domain adaptation, which includes a weighing method, innovatively calculated with data distributions by K-means clustering, to merge existing sites' information for new sites' power forecasting [13]. However, these efforts only used one certain clustering method and did not further explore other faster and more efficient clustering approaches.

As evidenced above, the effectiveness of cluster-based wind energy modeling analysis has been validated by multiple relevant models at wind sites worldwide, with engineering applicability and values. Nevertheless, except for the K-means algorithm, other well-suited clustering algorithms are rarely employed in this field. Interestingly, in this journal, Ref. [10] presented the significance of investigating different clustering approaches in wind power modeling. Wang et al. [14] conducted a self-organizing map clustering for classifying data and used neural networks and support vector machines as base learners to create a Bayesian model averaging ensembles for analyzing wind power. The model adapts to different meteorological conditions, but its clustering approach and learners are neural network-based and thus with high temporal complexity.

There remains a lack of comparative studies on ensemble learning

wind power modeling with different clustering algorithms. Nor is there existing research that combines various clustering approaches-based stacking ensembles and considers the data diversities introduced by clustering for the modeling tasks. Both of these gaps are addressed in this study.

## 1.2. Contribution

Drawing on the literature review above, this study focuses on a wind farm in the Norwegian Arctic. A wind power modeling framework is proposed, which involves quantifying wind turbulence, clustering meteorological data, and ensemble learning. Firstly, an effective model integrating bagging and boosting is constructed. Secondly, four prominent clustering algorithms are systematically incorporated with models to form layered cluster-based ensembles and the best clustering approach is selected. Finally, stacking is employed to fuse these ensembles with different clusters to establish a more accurate model.

The principal contributions of this paper are thus as follows.

1. This paper experimentally proves that farthest first clustering is a distinctive approach in clustering wind data for power modeling compared to K-means, expectation-maximization, and Canopy clustering algorithms. The paper shows that even the worst-performing layered cluster-based ensemble outperforms the one without clustering. This indicates the similarities and dissimilarities in wind data. However, even though these data are not related to an individual wind turbine, they can still be significantly reflected in wind power in an implicit form.
2. Given the differences in results of different clustering algorithms, the paper proposes fusing layered ensembles with varying clusters with two-layer stacking to formalize a model that exceeds the optimal single clustering method. The stacking can more efficiently and quickly address the complex mapping task of nonlinear relationships between meteorological wind data and wind power.
3. The paper builds a procedure for determining the cluster number with a heuristic elbow chart, an empirical formula, and an X-means clustering approach. The procedure may be further developed and refined into a technique for identifying cluster numbers on other problems.
4. AdaBoost boosting with random forest bagging as its weak learner is apposite in the wind power models. These tree-based algorithms are computationally fast and parameter insensitive compared to the network-based ones. The proposed AdaBoost model statistically outperforms linear, neural network, and benchmark Adaboost approaches.
5. The quantization of wind turbulence intensities—both wind speed and direction that are rarely considered in related research—is applied to wind power modeling in a novel manner. The study finds that both intensities can serve as new features for considering wind volatility in the modeling.

The remainder of this paper is organized as follows: Wind meteorology and the use of data are described in Section 2. In Section 3, an elaborated description of the clustering approaches and statistical methods is presented. Section 4 shows the experimental procedure. Section 5 presents and discusses the obtained results. Finally, Section 6 concludes this work, noting its implications and outlook.

## 2. Wind power meteorology and data preparation

### 2.1. Wind power

Wind power generation is the conversion of wind kinetic energy into electricity. Ignoring losses in the conversion process, the actual output power of wind turbines can be expressed as in (1):

$$P = \begin{cases} 0 & v < v_{min} \\ \frac{1}{2}C_p(v)\rho Av^3 & v_{min} < v < v_n \\ P_n & v_n < v < v_{max} \\ 0 & v > v_{max} \end{cases} \quad (1)$$

where  $P$  represents the output power of the wind turbine (W);  $C_p(v)$  represents wind energy utilization efficiency;  $\rho$  is the air density ( $\text{kg}/\text{m}^3$ );  $A$  represents the effective area swept by the wind turbine blades ( $\text{m}^2$ );  $v$  is the wind speed ( $\text{m}/\text{s}$ );  $v_{min}$ ,  $v_{max}$ , and  $v_n$  respectively represent cut-in, cut-off, and rated wind speeds; and  $P_n$  is the rated wind power for the wind turbine. From (1), it is clear that the output of a wind turbine is primarily influenced by wind speed, air density, and swept area. Moreover, air density is primarily affected by temperature and pressure [15]. The swept area is related to the wind direction.

## 2.2. Quantification of turbulence in the wind

Turbulence arises when airflow moves through uneven landscapes or differences in air density. Turbulence is an immensely complicated flow phenomenon that is highly stochastic and difficult to characterize. In actual wind farm operations, turbulence is generated because of topographic and climate conditions and weak effects between wind turbines. Turbulence has a particularly strong impact on wind power production: given similar wind speed conditions, the higher the turbulence intensity, the higher the impact on wind farm output power [16]. The wind turbine's large inertia includes an impeller, whose rotation is behind wind speed change. Therefore, the turbine will not get the theoretically predicted wind force, and the power output will go down. Empirically, at low wind speeds, turbulence increases turbine power production. However, when wind speed approaches the turbine's furling speed, turbulence reduces production [17]. Nevertheless, turbulence is rarely considered in machine learning models of wind energy. An article in the journal [18] compared the effects of five popular learning algorithms and nine atmospheric variables on wind turbine power generation and found the following through statistical tests. First, for the five benchmark algorithms, the selection of atmospheric features for wind power modeling is more important; second, the top five features that are most influential for modeling are, in order, wind speed, turbulent kinetic energy, temperature, turbulence intensity, and wind direction. However, turbulent kinetic energy is seldom recorded by wind sites due to its measurement complexity. Therefore, turbulence intensity is considered as an input feature in this study.

Turbulence intensity, defined as wind speed standard deviation divided by the mean value over a short period [19], is the principal characteristic quantity of wind speed volatility. The turbulence intensity of direction is also applied as a quantitative tool to define turbulence behavior in wind direction. The turbulence intensities are shown in (2):

$$I_{SP} = \frac{S_{SP}}{SP}, I_D = \frac{S_D}{D} \quad (2)$$

where  $I_{SP}$  and  $I_D$  are wind turbulence intensity of wind speed and direction;  $SP$  is wind speed;  $S_{sp}$  is its standard deviation of the previous 10 min;  $D$  is wind direction index;  $S_D$  is its period standard deviation.

## 2.3. Data preparation

The study centers on a 54 MW wind farm designed in northern Norway, located about 500 km inside the Arctic Circle; it stands out as one of the largest wind farms in the Arctic. This farm's terrain features are a small hill, high steep mountains, and fjords, which are regarded as complex terrains. The wind power station company offered measurement of wind data with 10 min temporal resolution. We chose the five-dimensional meteorological wind data (wind speed and its variance, wind direction and its variance, and temperature) and power data from

0:00 January 1, 2017 to 23:50 December 31, 2017. Specifically, we calculated the sine values of wind direction and its standard deviation as indicators of wind direction and its fluctuations. Further, the turbulence intensities of wind speed and sine value of direction were computed as quantitative indices of wind turbulence. In summary, 10-min resolution wind data—consisting of wind speed and sine direction and their turbulence intensities, temperature, and pressure—were employed to model wind power. These measurements contain small outliers and inevitable noise. However, the employed algorithm is insensitive to outliers, and this noise fits the standard normal distribution; therefore, they are not further considered. Because the scales of variables in the dataset vary widely, it is worth rescaling the original data into new data with similar proportions for each variable. Data normalization or standardization can increase model convergence speed and improve some algorithms' accuracy, especially in distance-based clustering [20].

## 3. Methodology

### 3.1. Chosen clustering approaches

Cluster analysis is an exploratory data mining technique for extracting useful information from high-dimensional datasets. It is a type of unsupervised learning approach to grouping similar hidden patterns [21] and classifying similar data into different subsets to give subset members identical attributes [22]. This classification requires quantifying the degree of similarity or dissimilarity between observations. The clustering results are strongly dependent on the kind of similarity metric used [23]. The cluster number is typically unknown and needs to be designated according to prior knowledge or determined by some method. Several clustering methods have been proposed. Ref. [24] offered some factors for choosing a clustering method. The method should be able to effectively and precisely find the suspected cluster types, and it must resist errors in the datasets; further, it must have the availability of computing power. This paper selects four clustering approaches: K-means, expectation-maximization, farthest first, and Canopy. The first one is a baseline method, and the other three can be regarded as competitors.

**K-means:** Among clustering algorithms, the K-means algorithm is one of the most popular and classical. It is a robust and versatile clustering algorithm proposed in Ref. [25]. The target of the K-means is to categorize observations into  $k$  clusters. K-means in this study is associated with Euclidean distance. Given a set of  $n$  data points  $D = \{\mathbf{x}_1, \dots, \mathbf{x}_n\}$  in  $\mathbb{R}^d$  and an integer  $k$ , the K-means problem is to determine a set of  $k$  centroids  $C = \{\mathbf{c}_1, \dots, \mathbf{c}_k\}$  in  $\mathbb{R}^d$  to minimize the following error function:

$$E(C) = \sum_{\mathbf{x} \in D} \min_{k=1, \dots, K} \|\mathbf{x} - \mathbf{c}_k\|^2 \quad (3)$$

It is a combinatorial optimization that equals finding the partition of the  $n$  instances in  $k$  clusters whose associated set of mass centers minimizes Eq. (3) [26].

**EM:** The Expectation-Maximization (EM) algorithm is proposed by Ref. [27]. It provides a simple, easy-to-implement, and efficient tool for the learning parameters of a model [28], and is widely used. It finds the maximum likelihood or maximum posterior of the parameters in a probabilistic modeling process where the model relies on latent unobservable variables. The EM first initializes distribution parameters, then alternates between two steps: computing the expectation of variables based on the assumed initial parameters; and maximization, which gives a maximum likelihood estimate of the current parameters through the expectation values of the latent variables. The two steps repeat iteratively until the desired convergence is realized. When applied to clustering, the probabilistic model is established on the probability of each data sample to each cluster and distributes samples to the cluster with the largest possibility. The goal of EM clustering is to maximize the

overall probability or likelihood of the clusters.

**FF:** The first utilization of Farthest First (FF) traversal is in Ref. [29]. FF is an effective greedy permutation method in computational geometry. Its underpinning is the traversal of a sequence of points in space where the initial point is specifically stochastic. The subsequent points are as remote as possible from the prior chosen set of points. FF clustering is the application of FF traversal in clustering, which was introduced in Ref. [30]. It is an optimized K-means with an analogous procedure, selecting the centroids first and assigning the samples to clusters with the maximum distance. Specifically,  $k$  numbers of centroids are generated by stochastically choosing a data point as the primary cluster centroid and greedily selecting the second centroid when it is FF from the first centroid. The process is repeated  $k$  times. As soon as all the centroids are recognized, FF assigns all the other data to the cluster in which the data have the nearest feature distance. In contrast to K-means, FF merely requires one traversal to cluster data. All the cluster centers are real data points, not geometric clustering centroids, and their position is fixed in the computation [31]. In most cases, the speed of clustering is considerably increased because fewer reassignments and adjustments are involved. The FF traversal is described in Algorithm 1.

#### Algorithm 1

Farthest First clustering Algorithm.

---

```

1. Farthest First Clustering (D: dataset, k: cluster number) {
2. select random data as the first point and first centroid;
3. // searching the data sample that is the farthest from the centroid
4. for (l=2,...,k) {
5. for (each remaining data sample in D) {
6. calculate the total distance to the existing centroids;}
7. select the sample with the largest distance as the new centroid;
8. label the centroids as {c1, c2, ..., ck}
9. //assignment the rest points {p1, p2, ..., pn}
10. for (each point pi) {
11. calculate the distance function dist to each fixed cluster centroid;
12. realize min {dist(pi, c1), dist (pi, c2), ..., dist (pi, ck)}
13. put it to the cluster with minimum distance;} }

```

---

**Canopy:** Canopy clustering was introduced in 2000, and its central idea was to use a cheap, approximate distance measure to divide the data into subsets efficiently. This clustering decreases computing time over K-means and EM clustering methods by more than an order of magnitude and reduces errors on large datasets [32]. Unlike K-means, which only uses one distance, the Canopy algorithm uses two threshold distances, the larger loose distance  $T_1$  and the smaller close distance  $T_2$ . It begins by removing a random point  $r$  sample from the original dataset and starting a *canopy* centered at  $r$ . It then approximates all distances between  $r$  and the remaining data  $r_i$ . If the distance is less than  $T_2$ , it places  $r_i$  in  $r$  canopy. If the distance is less than  $T_1$ , it removes  $r_i$  from a dataset. It repeats these steps until there is no more data to be clustered. However, Canopy needs to be tuned to the distance parameters and, according to Ref. [33],  $T_1$  and  $T_2$  can be obtained approximately using a heuristic based on attribute standard deviation.

### 3.2. Determining the cluster number

While various clustering methods are available, all of the mentioned methods require the cluster number before the clustering procedure. It is necessary to estimate the number due to the resulting partition of the data being dependent on its specification.

There have been energy studies using validated clustering methods, such as the elbow chart [10], Davies–Bouldin index [34], etc., to conduct data analysis. However, according to Refs. [35,36], there is no standard method for determining cluster numbers. Therefore, this paper combines three methods—formula, plotting, and information value—to comprehensively find a suitable cluster number for our meteorological wind data.

There is an empirical formula [37] to find the cluster number  $k$ . It is useful to check the range of  $k$  since it is not a precise approach.

$$k = 1 + 3.2 \log_{10} n \quad (4)$$

where  $n$  is the number of data points. This formula is inaccurate but can provide a reference for seeking  $k$ .

The elbow method is a visually heuristic technique for choosing cluster numbers [38]. The elbow principle's idea is that the total sum of squared errors between the sampling point in each cluster and the centroid (a smaller value means a more convergent result) is calculated with a series of  $k$  values. When the setup cluster number approximates the actual cluster number, the sum of squared errors will decrease swiftly. As the setup cluster number continues to grow, the Sum of Squared Errors (SSE) will continuously decrease, but more slowly [39]. Intuitive observation of turning points from elbow plots is sometimes vague. Still, it can provide a reasonable interval for searching for the value of  $k$ .

The X-means approach offers an effective tool for finding the exact  $k$ .

**X-means:** X-means is a variation of K-means clustering and can automatically determine the optimal cluster number in a dataset. It refines cluster assignment by repeatedly attempting subdivision segments and keeping the best resulting splits. It searches the space of cluster locations and the cluster number to optimize the Bayesian Information Criterion (BIC) measure [40]. The main parameters for X-means are the lower and upper bounds of the cluster number, which are found in the above two methods. It includes two steps that are repeated until they reach the required convergence. Primarily, the K-means algorithm is utilized to cluster the given dataset. Each cluster centroid is divided into two parts in opposite directions along a stochastic vector. The K-means algorithm is locally operated within the old cluster and generates two new clusters. By comparing the BIC scores of the original clustering structure with a new one, the splitting is either made or not. The idea is that splitting a single cluster into two clusters increases the BIC score, with two clusters being more probable than one. When  $k$  reaches the set upper bound, the splitting stops, and the algorithm reports BIC scores for each  $k$  value.

### 3.3. Modeling ensemble learning algorithm

The fundamental idea behind ensemble learning is to ensemble multiple algorithms or models to achieve an integrated model with better predictive performance [41]. The ensemble method can tactfully partition the dataset into smaller ones, train them separately, and then combine them with some strategies. The main strategies can be categorized into three groups: boosting, bootstrap aggregating (shortened as bagging), and stacking.

In the bagging procedure, new training sets are formed by taking from the original training set with a put-back. The averaging method for each new result of the training set is applied to reach the final result in a regression. Random forest [42] is an efficient bagging algorithm that uses decision trees as its base learners and offers decent performance and low computing costs. It is an improvement in the decision tree algorithm in which, essentially, multiple decision trees are merged. The creation of each tree depends on an independent bagging subset. Each tree in the forest has the same probability distribution. The final regression value can be determined by averaging each predictive value from each tree. Since random forest introduces perturbations in sampling and features, it dramatically improves generalization and avoids overfitting. Further, it can handle high-dimensional data without feature selection, and crucial features are derived during the training process [43].

Boosting [44] is an approach that boosts weak learners to strong learners. Adaptive boosting (shortened as AdaBoost) is a representative boosting algorithm. It continually builds weak learners to emphasize (with larger weights) on samples mislearned in the prior learner until the number of learners reaches the setup value or the loss function

reaches a threshold. For the regression problem, the weighted average is used to eventually obtain predicted values. Adaboost is highly accurate, can adequately construct weak learners, and is not susceptible to overfitting. Meanwhile, it is sensitive to anomalous samples (which may receive large weights in iterations), affecting the performance of strong learners. For the numeric output for the strong learner  $h_i(x) \in \mathbb{R}$ , weighted averaging (5) is used for the final result [45].

$$H(x) = \sum_{i=1}^M w_i h_i(x) \quad (5)$$

where  $w_i$  is the weight of a weak learner and  $w_i \geq 0, \sum_{i=1}^M w_i = 1$ .

Stacking is a representation learning technique that can extract valid features from data by employing meta-learning algorithms to learn how to optimally combine predictions from many base learners. Several different base models are first trained with the original dataset. A new model named meta-learner is then trained with each of the previous models' outputs to get a final output [46]. The stacking result is typically better than its single base learner since the fusional ensemble combines varying types of base learners. The applied stacking is shown in Algorithm 2 [47].

**Algorithm 2**

Stacking algorithm with four base learners and one meta-learner.

---

```

Input: Dataset  $D = \{(x_1, y_1), (x_2, y_2), \dots, (x_m, y_m)\}$ 
        Base learner varying clustering approaches-based Adaboost algorithms  $L_1, \dots, L_4$ ;
        Meta-learner linear regression  $L$ 
Process:
1. for  $t = 1, 2, 3, 4$  do
2.    $h_t = L_t(D)$ ;
3.   // Train base learners by  $L_t$ 
4. end for
5. // Generate training set for meta-learner
6.  $D' = \emptyset$ ;
7. for  $i = 1, 2, \dots, m$  do
8.   for  $t = 1, 2, 3, 4$  do
9.      $z_{it} = h_t(x_i)$ ;
10.  end for
11.   $D' = D' \cup \{(z_{i1}, z_{i2}, z_{i3}, z_{i4}), y_i\}$ 
12. end for
13. // Meta-learner  $h'$  is established
14.  $h' = L(D')$ 
Output:  $H(x) = h'(h_1(x), h_2(x), \dots, h_T(x))$ 
    
```

---

A two-layer assemblage structure for regression, which can be categorized as a kind of layered cluster-based or oriented ensemble named by Ref. [48], is adopted to optimally incorporate clustering results generated separately by the four above clustering approaches into the AdaBoost mechanism. The ensemble structure achieves excellent learning ability and prediction accuracy by mapping the first-layer clustering to the second-layer ensemble regression [21].

**3.4. Proposed modeling strategy**

A proposed framework for clustering approach comparisons is displayed in Fig. 1. It is inspired by wind energy meteorology, clustering approaches, and ensemble learning. The framework is a two-layer architecture, with four clustering algorithms in layer 1 and AdaBoost in layer 2. Specifically, the random forest is the weak learner for the AdaBoost, and Reduced-Error Pruning TREE (REPTREE) [49] is introduced to replace the decision tree in the random forest to reduce overfitting that may be caused by the complicated ensemble model structure.

Take K-means clustering as an example. Layer 1 uses Section 3.2 to identify the cluster number  $k$  and clusters the wind data into  $k$  clusters. Layer 2 employs each cluster to train the AdaBoost to learn AdaBoost and establish  $k$  submodels with labels. Subsequently, the test data, one by one, are classified into an existing cluster and loaded into the trained AdaBoost submodel corresponding to the cluster for wind power modeling, and overall performance is calculated.

Analogously, the above procedure is also applied to EM, FF, and Canopy clustering approaches. More experimental details are presented in Section 4.

Regarding stacking ensemble modeling, a novel method is put forward. It also consists of two layers, the first being base learners (AdaBoost models with the four different clustering algorithms) and the second being linear regression Eq.(6) with Tikhonov regularization  $\lambda \|w\|_1$  (also named ridge regression [49] Eq.(7) to avoid overfitting caused by the complex model structure [50]). The reasons for this configuration are the following. First, the first layer has diversity in the layered ensembles based on four clustering algorithms and may deeply extract data features and transmit them to the second layer. Second, the major risk of the second layer is that it learns the generated data from the first layer and is vulnerable to overfitting, so linear regression with a regular term

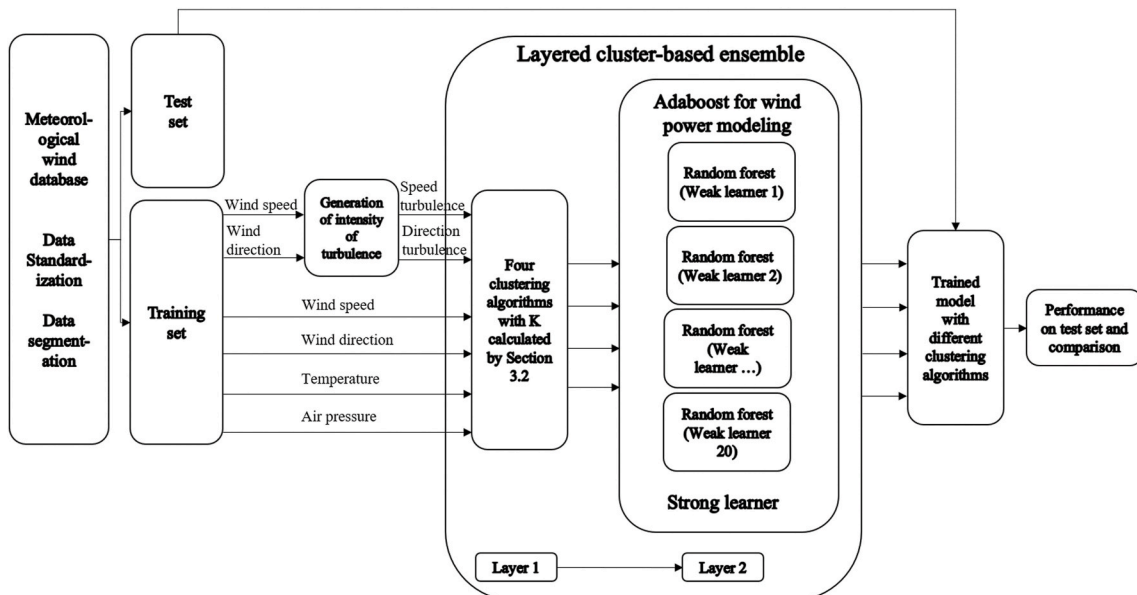


Fig. 1. The procedure of the proposed strategy for wind power modeling.

is the learning algorithm in this layer. The first layer procedure is the same as in Fig. 1. It generates four sets of simulated power on training and test sets. Subsequently, the second layer uses the measured power and four generated power sets as the dependent and independent variables, respectively, to build ridge regression on the training set and employs the learned regression to predict the power with the simulated test power on the test sets.

$$f(\mathbf{x}) = \mathbf{w}^\top \mathbf{x} + b \quad (6)$$

with a loss function  $J = \frac{1}{n} \sum_{i=1}^n (f(\mathbf{x}_i) - y_i)^2 + \lambda \|\mathbf{w}\|_1$  :

$$\min_{\mathbf{w}, b} \frac{1}{n} \sum_{i=1}^n (\mathbf{w}^\top \mathbf{x}_i + b - y_i)^2 \quad (7)$$

$$s.t. \quad \|\mathbf{w}\|_1 \leq t$$

### 3.5. Model evaluation metrics and multiple comparisons

Two metrics are utilized in evaluating the performance of different models in the test set. The first one is Normalized Mean Absolute Error (NMAE), while the second is Normalized Root Mean Square Error (NRMSE). They are negatively oriented, which means the smaller value is related to better performance. The NRMSE assigns a higher weight to larger errors because of the square calculation, meaning it punishes substantial prediction errors and reveals whether the regression has noticeable error variance.

$$NMAE = \frac{\sum_{i=1}^n |prediction_i - observation_i|}{n} \bigg/ \frac{\sum_{i=1}^n observation_i}{n} \quad (8)$$

$$NRMSE = \sqrt{\frac{\sum_{i=1}^n (prediction_i - observation_i)^2}{n}} \bigg/ \frac{\sum_{i=1}^n observation_i}{n} \quad (9)$$

Two statistical approaches are used to check whether there are statistically significant differences between the model's performance. The Friedman test is used to check for differences in performance across multiple trials [51]. It tests column effects after adjusting for possible row effects.

$H_0$ : The column data do not have a significant difference.

$H_a$ : They have a significant difference.

Its statistic  $F$  is shown as:

$$F = \frac{12n}{k(k+1)} \left[ \sum_{i=1}^k r_i^2 - \frac{k(k+1)^2}{4} \right] \quad (10)$$

where  $k$  is the number of columns,  $r_i$  is the mean value of row  $i$ , which follows  $\chi^2_{(k-1)}$  under  $H_0$ .

Furthermore, the Tukey method is used for computing confidence intervals between the means of two populations. It is expressed as follows:

$$(\bar{Y}_1 - \bar{Y}_2) \pm \frac{q_{k,n-k,1-\alpha}}{\sqrt{2}} \cdot \sqrt{MSE} \cdot \sqrt{\frac{1}{n_1} + \frac{1}{n_2}} \quad (11)$$

where  $q$  is the Gaussian  $q$ -distribution,  $k$  is the number of populations, and  $n$  is its total size;  $MSE$  is the Mean Square Error within groups.

## 4. Experiment setup

This study extracts meteorological wind data from the Norwegian Water Resources and Energy Directorate, including a few abnormal negative values, at which the wind farm did not generate electricity but consumed grid power. All weather data are normalized as inputs to the models. First, the wind data are divided into a training set, accounting for 90%, and a test set, accounting for 10%. To fully apply the data, avoid overfitting, and improve generalization in modeling [52], 10-fold

cross-validation is used in the training. Then, weather data are harnessed in the test set to calculate the corresponding wind power, which is compared to the actual power data to obtain performance metrics.

For the benchmark model, the processed training data are directly fed into the AdaBoost with random forest (Layer 2 in Fig. 1) (AdaRF). The number of iterations is set to 100 (trade-off between performance and computing speed). Random forest is the AdaBoost inner weak learners, and the number of REPTREE in each random forest is set to 10. The competitors are linear regression (LR), artificial three-layer neural networks (16 nodes in the hidden layer, which is found by a grid search from 6 to 20) (ANN), and AdaBoost with 20 decision trees (achieved by a grid search from 5 to 30 with an interval of 5) as its weak learners (AdaDT).

Regarding the ensemble model based on clustering approaches, the range of cluster numbers for the weather data is first found by the elbow graph and empirical formula (4). Its exact value is determined using the X-means clustering method. Then, the four aforementioned clustering approaches are used to group the data in the training set and categorize the test data into established clusters to find the best-performing clustering algorithm. Finally, stacking is employed to combine layered cluster-based ensembles with different clustering algorithms to further explore avenues to upgrade power modeling.

In this study, wind power modeling is realizing the relationship between wind power and wind weather  $\mathbf{W}_t$ . The model is shown in (12).

$$\hat{P}_t = f_t(\mathbf{W}_t) + e \quad (12)$$

in which

$$\mathbf{W} = [V, IV_{turbulence}, \sin(\theta), I\sin_{turbulence}(\theta), T, P] \quad (13)$$

where  $\hat{P}_t$  is modeling wind power;  $f_t(\cdot)$  is the model that needs to be implicitly realized;  $\mathbf{W}_t$  represents weather data that will be clustered by the four clustering approaches;  $e$  is the model error.

## 5. Experiments and results

### 5.1. Feature ranking and comparison for modeling without clustering

The training wind data are initially harnessed to establish a multivariate linear wind power regression model to check the feature attributing degree. The diagnosis ( $T$  statistic and its corresponding two-tailed  $p$ -value [53]) for the interpretation of each feature is shown in Table 1.

All meteorological features are statistically significant in the linear modeling except pressure  $P$ . The features' importance may be approximately ranked by absolute values of  $T$  statistics in a descending scale as  $V, \sin(\theta), T, IV_{turbulence}, I\sin_{turbulence}(\theta), P$ . Although pressure does not contribute to the linear regression, all above meteorological features are still accounted for in the modeling as pressure values are relatively stable and the presented models are clustering and tree models demanding low computations and feature selection.

To enhance the verifiability of modeling results, the year is split into four quarters, Q1, Q2, Q3, and Q4, for individual power modeling. The statistical variability among quarterly data is initially analyzed in Table 2. Statistics and distribution disparities between meteorological wind and power quarterly datasets can be summarized, and quarterly data differ from yearly data. Therefore, separate modeling on these datasets can strengthen the proposed strategy's credibility.

The four quarterly and yearly normalized training data are separately entered into the proposed AdaRF, benchmarking LR, ANN, and AdaDT to map the relationship between wind data and wind power. Fig. 2 shows the results. Both NMAE and NRMSE increase significantly as time grows. The NMAE and NRMSE of AdaRF are significantly lower than the results obtained from multivariate linear regression. The average NMAE and NRMSE decrease by 52.98% and 46.31%,

**Table 1**  
The wind features were selected by the statistical diagnosis of linear regression.

Futures	V	$IV_{turbulence}$	$Sin(\theta)$	$ISin_{turbulence}(\theta)$	T	P
T statistic; p-value	250.79; <0.0001	7.84; <0.0001	-21.77; <0.0001	3.02; 0.0025	23.68; <0.0001	0.67; 0.5040

Note: the term is shown as “T statistic; p-value.” The  $H_0$  is where the interpretation equals zero and its  $H_a$  is where the term is not zero; when the p-value is smaller than the set confidence level of 0.05, the  $H_0$  is rejected and the feature is attributed to the linear model.

**Table 2**  
The statistics of the yearly and quarterly wind data.

Dataset	Statistics			
	Average	Standard deviation	Skewness	Kurtosis
Year	<0.0001	0.9983	4.9493	100.7258
Q1	0.1988	0.8981	3.9305	87.7615
Q2	-0.0283	0.8729	5.4827	129.5278
Q3	-0.0356	0.8421	5.4604	118.6208
Q4	-0.1350	0.8873	4.7543	113.6719
CoV for Statistics	-53584	0.0586	0.1157	0.1316

Note: The different variables are standardized to similar scales, so the statistics of the various variables in the dataset are averaged and shown. Coefficient of Variation (CoV) is defined as the ratio of standard deviation to mean.

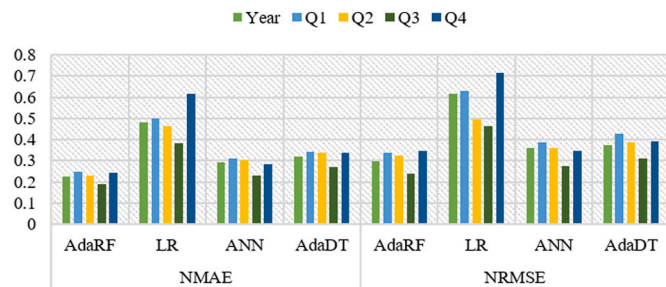


Fig. 2. The performance comparison for power modeling without clustering.

respectively. The AdaRF decrease in NMAE and NRMSE (corresponding to the model improvement) is also evident when compared with ANN (NMAE 19.54% and NRMSE 10.43%) and AdaDT (NMAE 29.31% and NRMSE 17.95%). Fig. 3 displays the modeling power of a day, from which AdaRF appears close to real values but with several errors in points. This means the proposed AdaRF enables accurate power modeling based on weather data but still leaves room for refinement.

5.2. Determination of an appropriate cluster number

The size of the yearly dataset is 52,560;  $k$  is calculated to be approximately 16 in (4). Selecting this value as the midpoint, the total Sum of Squared Errors (SSE) of K-means is calculated with a starting point of  $k$  equals 2 and an endpoint of  $k$  equals 30. The elbow plot is drawn and displayed in Fig. 4. The precise value of the elbow point for the total sum of squared errors can be determined only approximately

since the process is by intuition and experience. However, Fig. 4 still shows an interval, the cluster number  $k \in [10, 20]$ , in which the decline of SSE begins to flatten from steep, and where the elbow point belongs to.

To demonstratively find the precise value of  $k$ , an X-means approach is adopted. The lower and upper bounds of  $k$  are set as 10 and 20 respectively according to the interval formerly found in Fig. 4. The optimized BIC score is 69,956.78 with a proper  $k$  value for the meteorological wind data equaling 11.

Analogously, the  $k$  values for the four quarterly wind data, Q1, Q2, Q3, and Q4, are decided as 14, 8, 9, and 12, respectively.

5.3. Comparison of different clustering approaches in modeling

For the yearly dataset, the four clustering approaches yield varying numbers of samples per cluster, albeit at the same cluster number. Based on the four separate clustering methods separately with 11 clusters, four complete layered cluster-based ensembles are developed for clustering comparisons. Firstly, the wind weather data with different clustering approaches in the training and test sets are shown in Fig. 5. Each color represents a cluster, and the vertical axis shows the percentage of each number in the total dataset.

The various clustering methods produce wildly different clustering results, even with the same  $k$ . The cluster sample number variance analysis reveals that the K-means method produces more homogeneous clustering than other methods. Even single-digit sample percentages are seen in the FF and Canopy algorithms for the training set. Apart from the K-means, all of the other three algorithms generate clusters that exceed one-fifth of the sample size. Second, the four clustering methods' yearly meteorological wind test data are loaded into the layered cluster-based ensemble for wind power modeling. The NMAE and NRMSE are displayed in Fig. 6.

The model based on FF clustering intuitively presents the smallest NMAE and NRMSE, 32.35% and 33.64% reduction without clustering; the second smallest is the model with Canopy. The models with these two clustering approaches significantly improve their performance compared to the ones without clustering, while the K-means and EM algorithms also upgrade their models' abilities.

To further elaborate comparisons of clustering methods, analogously, layered cluster-based ensemble modeling with different clustering approaches is conducted on four quarterly wind datasets, and their NMAE and NRMSE are displayed in Fig. 7. The ranking of the models built on quarterly data is the same as those built on yearly data. Strengths in the FF and Canopy clustering algorithms are evident in each quarter. Moreover, a result is derived that the 3rd quarter-power model performs the best, followed by the 2nd quarter. This illustrates a more

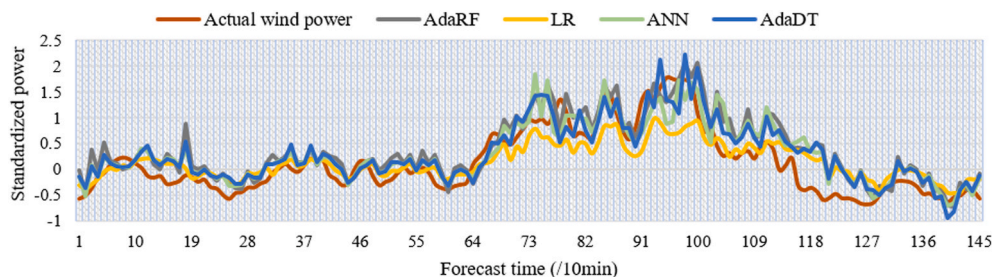


Fig. 3. The wind power modeling from weather data in the wind farm.

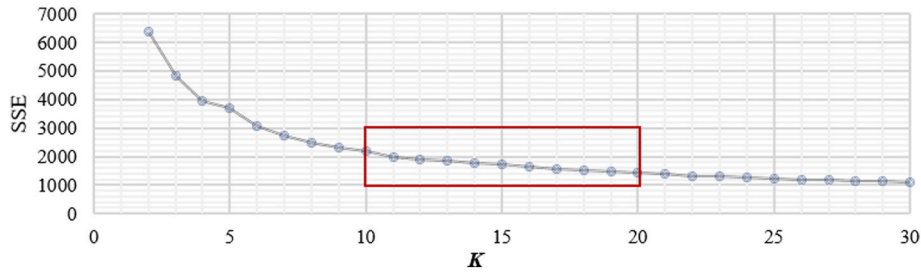


Fig. 4. The elbow plot for finding cluster number  $k$ .

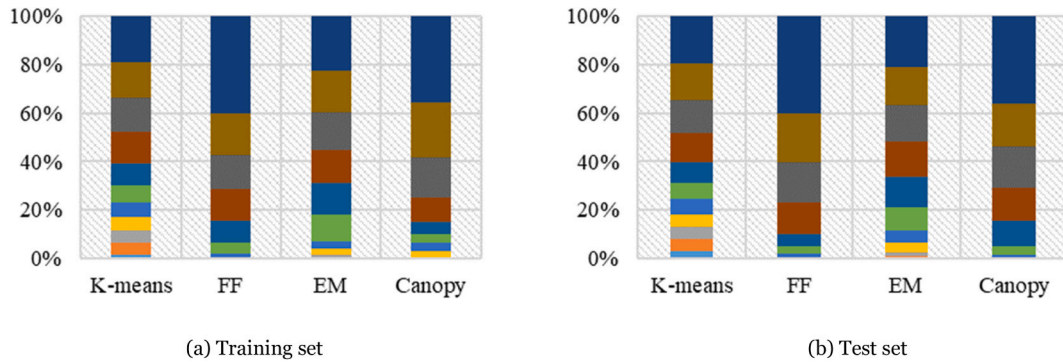


Fig. 5. The clusters number percentage of different clustering approaches.

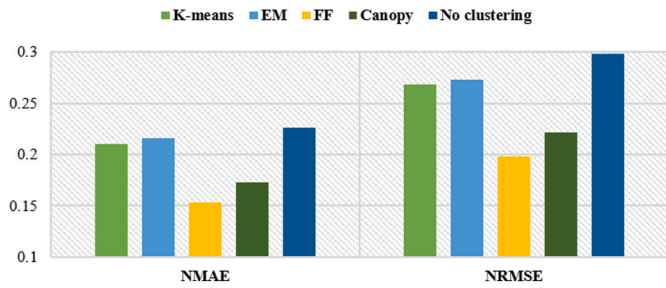


Fig. 6. The NMAE and NRMSE of the power modes with different clustering methods for the yearly dataset.

clear relationship between wind data and power from April to September, which is consistent with the intuition that the area has milder weather during this period.

Multiple comparisons are conducted between the metrics from different cluster-based models between quarters. The Friedman test  $p$ -values of NMAE and NRMSE are both 0.0056 and much smaller than the confidence level of 0.05, so the null hypotheses are rejected. This leads

to the conclusion that there are differences between the metrics of the various ensembles.

Table 3 compares the average NMAE and NRMSE for models with different kinds of clustering against the ones without. Quarterly evenly, the new model reduces NMAE and NRMSE by 13.94% and 17.45%. Furthermore, performance improvement between the two models generally slumps from summer to winter.

Regarding the best-performing FF clustering of the yearly dataset, Table 4 compares NMAE and NRMSE for the FF-based model to the original model. Both NMAE and NRMSE have an approximately 23%–34% decrease, which indicates this clustering approach is twice as good as the average clustering method in our case. Further, the superiority of FF is different with the quarter: the model boosting results are more

Table 3

The average performance improvement between the models with clustering and the corresponding one without.

	Q1	Q2	Q3	Q4
NMAE	13.66%	15.06%	16.29%	10.74%
NRMSE	17.73%	19.60%	20.08%	12.37%

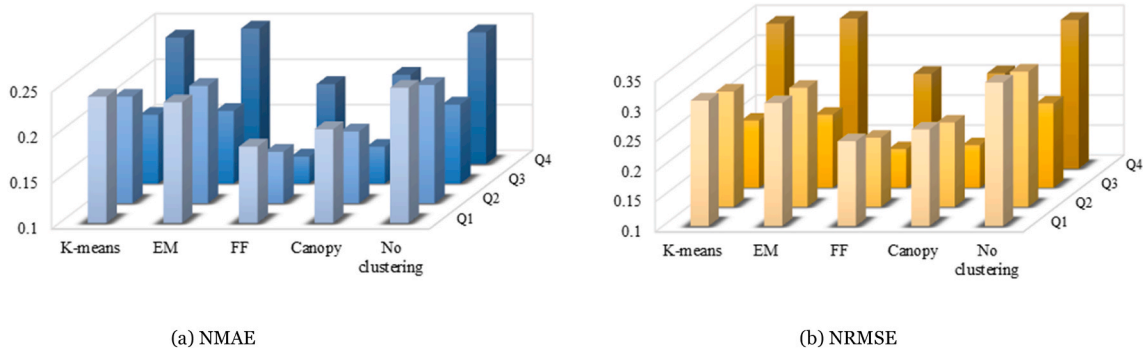


Fig. 7. The NMAE and NRMSE of the power models with different clustering methods for quarterly data.



**Table 4**

The performance improvement between the models with FF clustering and the baseline.

	Q1	Q2	Q3	Q4
NMAE	25.98%	31.94%	30.60%	23.24%
NRMSE	28.83%	33.85%	31.68%	25.88%

noticeable during warm periods compared to those during cold seasons.

Further, the Canopy clustering-based approach also displays a satisfactory result, which improves the modeling performance by averages of over 20% in NMAE and 25% in NRMSE, respectively. EM and K-means clustering-based approaches have relatively similar performance. However, they are still not as good as the FF and Canopy, as the above yearly analysis shows.

Collectively, the Tukey method calculates the intervals with 95% confidence of metrics difference; Table 5 shows the bounds of these intervals between the yearly and quarterly models with no clustering and ones with varying clustering algorithms. The upper and lower bounds of the metrics difference between no and FF clustering are positive, indicating that the superiority of the FF is statistically significant across multiple datasets. Moreover, the upper bounds of all other differences are greater than the lower bounds of absolute values, illustrating that normally distributed differences have positive means, which describes the other cluster-based models as outperforming no clustering in a probabilistic sense. Therefore, the edges of wind data clustering, ranking as FF, Canopy, K-means, and EM in order, before the layered ensemble modeling procedure, are demonstrated in our datasets.

#### 5.4. Stacking ensemble power modeling

Section 5.1 demonstrates that the proposed AdaRF outperforms three other benchmarks (ANN, AdaDT, and LR in descending order). Section 5.3 illustrates in Fig. 5 the four clustering algorithms yielding highly diverse clusters, and the cluster-based models work better. The AdaRF model is further refined by implementing a two-layer stacking structure (Cls-AdaRF): The first layer takes the four clustering outcomes in Fig. 5 as inputs to AdaRF to generate four-layered cluster-based ensembles; the second layer combines these ensembles outputs by linear regression to yield final simulations. Its performance is compared with that of AdaRF without clustering (NCl-AdaRF) in Section 5.1 and AdaRF with FF clustering (FF-AdaRF) in Section 5.3. The comparison in Fig. 8 shows that Cls-AdaRF decreases more NMAE and NRMSE in percentage than NCl-AdaRF (NMAE 31.89% and NRMSE 34.74%) and FF-AdaRF (NMAE 4.32% and NRMSE 5.71%). Its edge over FF-AdaRF indicates that stacking combined with four different clustering algorithms outperforms the best-layered ensemble with a single clustering. These model quarterly variations are similar to those in Section 5.3.

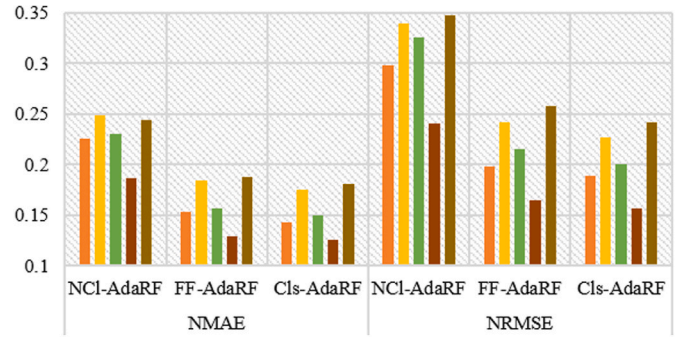
Likewise, placing benchmark algorithms ANN, AdaDT, and LR into this process eventually yields three cluster-based stacking ensembles, denoted as Cls-LR, Cls-ANN, and Cls-AdaDT. These models, including those that are run on the datasets separately, and their NMAE and NRMSE, are compared to those of Cls-AdaRF. Table 6 shows the difference intervals calculated by the Tukey method and reveals that, except for Cls-AdaRF vs. Cls-ANN in NMAE (where the upper bound is considerably close to zero), all the intervals are negative, indicating a

**Table 5**

The bounds for paired comparisons of clustering across yearly and quarterly datasets.

No clustering	v.s.	K-means	EM	FF	Canopy
NMAE	Lower Bound	-0.0363	-0.0414	0.0176	0.0011
	Upper Bound	0.0585	0.0534	0.1124	0.0959
NRMSE	Lower Bound	-0.0572	-0.0642	0.0122	-0.0030
	Upper Bound	0.1082	0.1012	0.1776	0.1624

Year Q1 Q2 Q3 Q4



**Fig. 8.** The NMAE and NRMSE of the power models with different clustering methods for quarterly data.

**Table 6**

The bounds for paired comparisons of stacking across yearly and quarterly datasets.

Cls-AdaRF	v.s.	Cls-LR	Cls-ANN	Cls-AdaDT
NMAE	Lower Bound	-0.3735	-0.1398	-0.1654
	Upper Bound	-0.2154	0.0183	-0.0073
NRMSE	Lower Bound	-0.3896	-0.1835	-0.2011
	Upper Bound	-0.2097	-0.0036	-0.0212

95% statistical significance among datasets for the Cls-AdaRF model's strength.

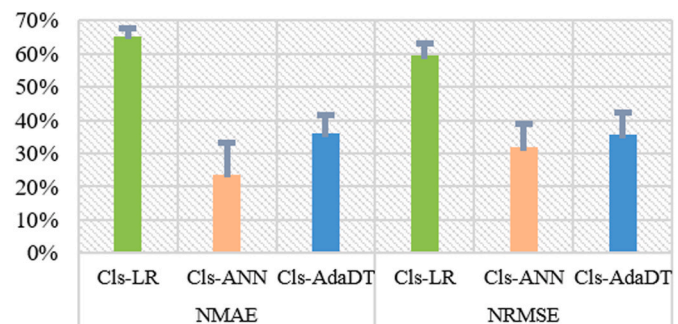
The percentage reductions in NMAE and NRMSE for Cls-AdaRF versus other models (corresponding to model improvement) are further calculated and presented in Fig. 9. On average, Cls-AdaRF delivers over 20% improvements over its three competitors (within a standard deviation; Cls-AdaRF outperforms Cls-LR, Cls-ANN, and Cls-AdaDT by about 60%, 25%, and 35%, respectively.).

Altogether, the Cls-AdaRF model is inferred to be a superior wind power model because it not only outperforms AdaRF without clustering but is also better than other stacking models.

## 6. Conclusions

This paper presents an ensemble learning approach that combines bagging, boosting, and stacking for modeling wind power from meteorological data. To mine the inherent characteristics of the data, four clustering approaches are used to process inputs for the layered ensembles. Then, the layered cluster-based ensembles are fused within the stacking framework. The proposed models' superiority is verified by diverse comparisons.

The AdaRF can accurately model wind power. The algorithm circumvents issues of an equal weighting of each tree in RF and AdaBoost



**Fig. 9.** The NMAE and NRMSE improvement of Cls-AdaRF versus other models for the yearly dataset.

and allows each learner to boost incrementally, and eventually creates a model with a good generalization. The overall performance of the proposed method is on average 33.94% in NMAE and 24.90% in NRMSE, lower as compared to the benchmarks in the cases excluding clustering.

As no standard methods for identifying the cluster number exist, the paper uses a heuristic elbow graph, an empirical formula, and the X-means clustering algorithm to precisely determine the implied number for meteorological data. Interestingly, the number for the yearly dataset is 11, which is close to the month's number. This result suggests that there may be analogous phenomena to measured wind data with monthly periodicity.

A comparative study of AdaRF based on different clustering methods reveals, firstly, that the model with clusters significantly performs better than the model without, regardless of what clustering approach is employed. This suggests that similarities within the wind power data can correspond to similarities within the weather data. Secondly, among these clustering methods, the model with FF clustering provides the best modeling results. The reason is that FF is built on finding the data point furthest from the previous centroid as the new one; in other words, it emphasizes large differences between clusters. Upon this clustering, the fluctuations among the original meteorological data are considerably diminished, which in turn corresponds to a smoother wind power output and increases the accuracy of the wind power model. The fast computability and accuracy of FF also suggest that the clustering technique can be applied to ultra-short-term wind power models. Thirdly, Canopy is the fastest among the four clustering methods and achieves comparable results. Therefore, Canopy can also serve as a favorable clustering approach when wind weather datasets are considerably large.

Finally, the wind power model is further strengthened by using stacking Cls-AdaRF to fuse the layered ensembles with four clustering approaches. It can be interpreted as Cls-AdaRF working as a representation learning—that is, effective features are automatically collected from raw data and fed into the second layer via multiple learners in the first layer; the second layer compiles and aggregates these features through linear regression with a regular term and effectively outputs simulations.

Practically, given that the proposed well-performing wind power model does not involve complex network training and extensive parameter tuning and wind energy physical modeling, it can be easily transferred to other energy utilization scenarios.

Further research, as suggested by the above conclusions, is needed to deeply optimize the base learners of stacking and their combination algorithms to deliver faster and more accurate modeling. Another direction is to incorporate this article's in-the-now power modeling approach and meteorological data with historical wind power to achieve efficacious short-term power forecasting.

#### Credit author statement

**Hao Chen:** Conceptualization, Methodology, Formal analysis, Visualization, Investigation, Writing, and Revise.

#### Data availability

Data for this study are available from the corresponding author provided reasonable requirements are present.

#### Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

#### Acknowledgments

The author thanks Dr. Yngve Birkelund for organizing the data and

comments, Dr. Xiuhua Zhu's help in the author's visiting the Max Planck Institute for Meteorology, during which the article was revised, and Dr. Bjørn-Morten Batalden, Dr. Abbas Barabadi, and Gunn-Helene Turi for their pertinent comments and valuable support. The publication charge has been funded by a grant from the publication fund of UiT The Arctic University of Norway.

#### References

- [1] Lee J, et al. Global wind report 2019. Brussels: Global Wind Energy Council (GWEC); 2020.
- [2] Tian Z, Ren Y, Wang G. Short-term wind power prediction based on empirical mode decomposition and improved extreme learning machine. *Journal of Electrical Engineering & Technology* 2018;13(5):1841–51.
- [3] Marvuglia A, Messineo A. Monitoring of wind farms' power curves using machine learning techniques. *Appl Energy* 2012;98:574–83.
- [4] Foley AM, Leahy PG, Marvuglia A, McKeogh EJ. Current methods and advances in forecasting of wind power generation. *Renew Energy* 2012;37(1):1–8.
- [5] Liu R, Peng M, Xiao X. Ultra-short-term wind power prediction based on multivariate phase space reconstruction and multivariate linear regression. *Energies* 2018;11(10):2763.
- [6] Ma J, Yang M, Han X, Li Z. Ultra-short-term wind generation forecast based on multivariate empirical dynamic modeling. *IEEE Trans Ind Appl* 2017;54(2):1029–38.
- [7] Niu D, Pu D, Dai S. Ultra-short-term wind-power forecasting based on the weighted random forest optimized by the niche immune lion algorithm. *Energies* 2018;11(5):1098.
- [8] Dong Y, Zhang H, Wang C, Zhou X. Wind power forecasting based on stacking ensemble model, decomposition and intelligent optimization algorithm. *Neurocomputing* 2021;462:169–84.
- [9] Kushwah V, Wadhvani R, Kushwah AK. Trend-based time series data clustering for wind speed forecasting. *Wind Engineering*; 2020. 0309524X20941180.
- [10] Dong L, Wang L, Khahro SF, Gao S, Liao X. Wind power day-ahead prediction with cluster analysis of NWP. *Renew Sustain Energy Rev* 2016;60:1206–12.
- [11] Wang K, Qi X, Liu H, Song J. Deep belief network based k-means cluster approach for short-term wind power forecasting. *Energy* 2018;165:840–52.
- [12] Tasnim S, Rahman A, Oo AMT, Haque ME. Wind power prediction using cluster based ensemble regression. *Int J Comput Intell Appl* 2017;16(4):1750026.
- [13] Tasnim S, Rahman A, Oo AMT, Haque ME. Wind power prediction in new stations based on knowledge of existing Stations: a cluster based multi source domain adaptation approach. *Knowl Base Syst* 2018;145:15–24.
- [14] Wang G, Jia R, Liu J, Zhang H. A hybrid wind power forecasting approach based on Bayesian model averaging and ensemble learning. *Renew Energy* 2020;145:2426–34.
- [15] Jung C, Schindler D. The role of air density in wind energy assessment—A case study from Germany. *Energy* 2019;171:385–92.
- [16] Kaiser K, Langreder W, Hohlen H, Højstrup J. Turbulence correction for power curves. In: *Wind energy*. Springer; 2007. p. 159–62.
- [17] Lubitz WD. Impact of ambient turbulence on performance of a small wind turbine. *Renew Energy* 2014;61:69–73.
- [18] Optis M, Perr-Sauer J. The importance of atmospheric turbulence and stability in machine-learning models of wind farm power production. *Renew Sustain Energy Rev* 2019;112:27–41.
- [19] Türk M, Emeis S. The dependence of offshore turbulence intensity on wind speed. *J Wind Eng Ind Aerod* 2010;98(8–9):466–71.
- [20] Ng A. Advice for applying machine learning. In: *Machine learning*; 2011. p. 3–15.
- [21] Verma B, Rahman A. Cluster-oriented ensemble classifier: impact of multicluster characterization on ensemble classifier learning. *IEEE Trans Knowl Data Eng* 2011;24(4):605–18.
- [22] Berkhin P. A survey of clustering data mining techniques. In: *Grouping multidimensional data*. Springer; 2006. p. 25–71.
- [23] Serra A, Tagliaferri R. Unsupervised learning: clustering. In: *Encyclopedia of bioinformatics and computational biology*. Amsterdam, The Netherlands: Elsevier; 2019. p. 350–7.
- [24] Milligan GW. Clustering validation: results and implications for applied analyses. In: *Clustering and classification*. World Scientific; 1996. p. 341–75.
- [25] MacQueen J. Some methods for classification and analysis of multivariate observations. In: *Proceedings of the fifth Berkeley symposium on mathematical statistics and probability*, vol. 1; 1967. p. 281–97. no. 14: Oakland, CA, USA.
- [26] Capó M, Pérez A, Lozano JA. An efficient approximation to the K-means clustering for massive data. *Knowl Base Syst* 2017;117:56–69.
- [27] Dempster AP, Laird NM, Rubin DB. Maximum likelihood from incomplete data via the EM algorithm. *J Roy Stat Soc B* 1977;39(1):1–22.
- [28] Do CB, Batzoglou S. What is the expectation maximization algorithm? *Nat Biotechnol* 2008;26(8):897–9.
- [29] Rosenkrantz DJ, Stearns RE, Lewis I, Philip M. An analysis of several heuristics for the traveling salesman problem. *SIAM J Comput* 1977;6(3):563–81.
- [30] Hochbaum DS, Shmoys DB. A best possible heuristic for the k-center problem. *Math Oper Res* 1985;10(2):180–4.
- [31] Devi RDH, Bai A, Nagarajan N. A novel hybrid approach for diagnosing diabetes mellitus using farthest first and support vector machine algorithms. *Obesity Medicine* 2020;17:100152.

- [32] McCallum A, Nigam K, Ungar LH. Efficient clustering of high-dimensional data sets with application to reference matching. In: Proceedings of the sixth ACM SIGKDD international conference on Knowledge discovery and data mining; 2000. p. 169–78.
- [33] R. R. Bouckaert et al., "WEKA manual for version 3-9-1," University of Waikato, Hamilton, New Zealand, pp. 47-49 2016.
- [34] Li Y, Wang B, Yang Z, Li J, Chen C. Hierarchical stochastic scheduling of multi-community integrated energy systems in uncertain environments via stackelberg game. *Appl Energy* 2022;308:118392.
- [35] Kodinariya TM, Makwana PR. Review on determining number of cluster in K-means clustering. *Int J* 2013;1(6):90–5.
- [36] Sinaga KP, Yang M-S. Unsupervised K-means clustering algorithm. *IEEE Access* 2020;8:80716–27.
- [37] Braverman V, Meyerson A, Ostrovsky R, Roytman A, Shindler M, Tagiku B. Streaming k-means on well-clusterable data. In: Proceedings of the twenty-second annual ACM-SIAM symposium on Discrete Algorithms. SIAM; 2011. p. 26–40.
- [38] Ng A. Clustering with the k-means algorithm. *Machine Learning Course*; 2012.
- [39] Yuan C, Yang H. Research on K-value selection method of K-means clustering algorithm. *J-Multidisciplinary Scientific Journal* 2019;2(2):226–35.
- [40] Pelleg D, Moore AW. X-means: extending k-means with efficient estimation of the number of clusters. *Icml* 2000;1:727–34.
- [41] Rokach L. Ensemble-based classifiers. *Artif Intell Rev* 2010;33(1–2):1–39.
- [42] Breiman L. Random forests. *Mach Learn* 2001;45(1):5–32.
- [43] Fernández-Delgado M, Cernadas E, Barro S, Amorim D. Do we need hundreds of classifiers to solve real world classification problems? *J Mach Learn Res* 2014;15(1):3133–81.
- [44] Breiman L. Bias, variance, and arcing classifiers. In: Tech. Rep. 460. Berkeley: Statistics Department, University of California; 1996.
- [45] Bertoni A, Campadelli P, Parodi M. A boosting algorithm for regression. In: International conference on artificial neural networks. Springer; 1997. p. 343–8.
- [46] Breiman L. Stacked regressions. *Mach Learn* 1996;24(1):49–64.
- [47] Zhou Z-H. Ensemble learning. In: Machine learning. Springer; 2021. p. 181–210.
- [48] Rahman A, Verma B. Novel layered clustering-based approach for generating ensemble of classifiers. *IEEE Trans Neural Network* 2011;22(5):781–92.
- [49] McDonald GC. Ridge regression. *Wiley Interdisciplinary Reviews: Comput Stat* 2009;1(1):93–100.
- [50] Džeroski S, Ženko B. Is combining classifiers with stacking better than selecting the best one? *Mach Learn* 2004;54(3):255–73.
- [51] Gibbons JD, Chakraborti S. Nonparametric statistical inference: revised and expanded. CRC press; 2014. p. 353–93.
- [52] Yadav S, Shukla S. Analysis of k-fold cross-validation over hold-out validation on colossal datasets for quality classification. In: 2016 IEEE 6th International conference on advanced computing (IACC). IEEE; 2016. p. 78–83.
- [53] Seber GA, Lee AJ. Linear regression analysis. John Wiley & Sons; 2012. p. 265–327.